# HW3 心得体会

## 心得体会：

带着几乎对作业完全不知如何下手的窘境，无奈下，不得不将别人的代码输入。但是抱着一定要将代码弄懂的心态，开始了我的学习之旅：

关于代码（三个部分）：

数据预处理：说到底，就是将句子转换为(tokenization)单字；

利用google开发的word2vec进行字嵌入，这个过程实际上只需要将语料库丢如Word2Vec() function即可，可是因为一开始什么都不懂，所以搜寻了大量资料来了解word embedding的原理，当然在这个过程中学习到了Word2Vec function的parameters各自代表什么含义，为后来训练词矢量模型奠定了基础；

利用对RNN模型进行训练并对文本进行分类，同样因为不了解，所以去硬着头皮学习RNN的建模原理，虽然过程会有点痛苦，可是又有收获——对Long Short-term Memory有了进一步了解；

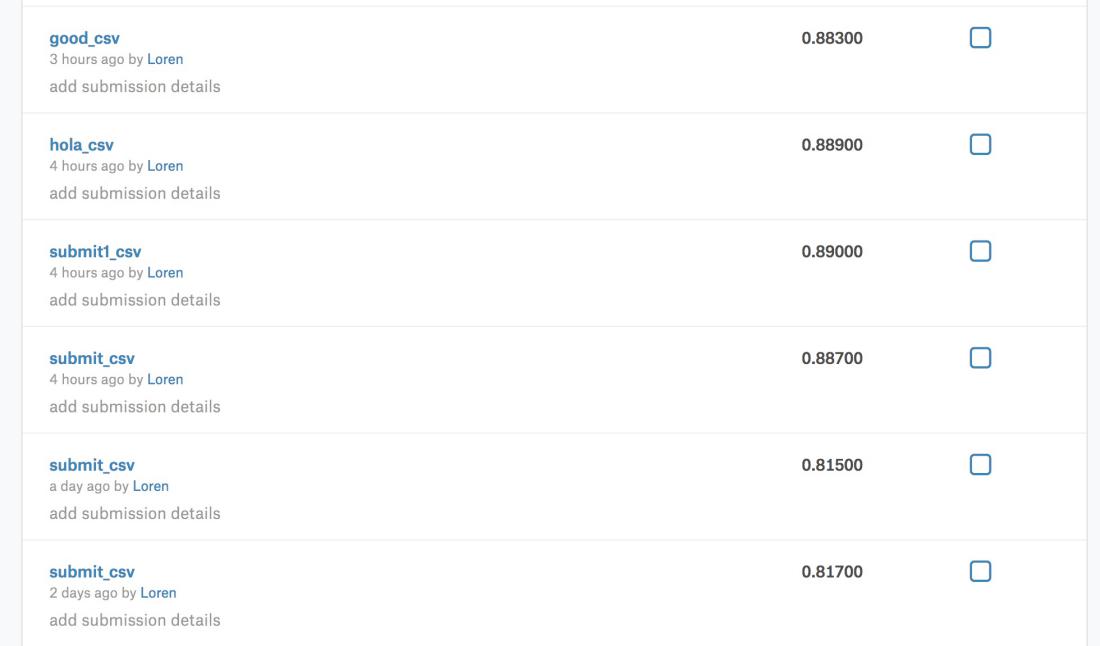
关于训练过程：

关于词矢量的训练，一开始全是default value, 结果是82 percent左右；

可是在我将Word2Vec() function中的parameters进行一系列调整——size = 250, iter = 10, window = 8, min\_count = 1, sg = 1，发现准确率跃升至88.7 percent；

紧接着，我想在最后的训练模型中多加一层，在选择activation function时，我本来想尝试运用Leaky Relu，无奈无功而返，原因还需要向助教请教。索性用了Relu，考虑到Sigmoid 跟 TanH存在梯度消失问题，故舍弃，可喜的是，准确率升至89 percent；

之后尝试将size缩小至220，希望可以紧凑矢量，避免稀疏，同时过大window的value，不知是否因为overfitting的原因，效果都没有更好。



透过作业，自己对RNN以及word embedding有了相对全面和深入的了解，从原来在各种资料间像无头苍蝇一样抓不住重点到现在可以捋出一条思路，并且对之前看过的资料做了整理及分类和提取，收获很大。

# 资料总结：

2018.5.6RNN分享学习笔记

* 什么是梯度消失问题？
* 梯度消失会引起什么样的后果？
* <http://mropengate.blogspot.tw/2017/02/deep-learning-role-of-activation.html> 使用激励函数的目的，如何选择激励函数
* <https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6> 对Neural Network中activation function的讨论
* <https://chtseng.wordpress.com/2017/07/24/neural-networks-%E4%B8%80/> 讲解了NN，即Neural Network的分类：￼

了解word embedding

* <https://machinelearningmastery.com/develop-word-embeddings-python-gensim/> 如何利用python提供的Gensim进行word embedding. “In this tutorial, we are going to look at how to use two different word embedding methods called word2vec by researchers at Google and GloVe by researchers at Stanford.”也就是说，word2vec跟Glove是并列的[告诉你如何实践]
* <https://blog.csdn.net/thriving_fcl/article/details/51406780> 动手训练一个词矢量空间（现在看来通俗易懂）
* <http://zake7749.github.io/2016/08/28/word2vec-with-gensim/> 以gensim训练一个中文词矢量
* <https://www.youtube.com/watch?v=X7PH3NuYW0Q&feature=youtu.be> 李宏毅讲word embedding 的原理：涉及1-of-N encoding；CBOW(predicting the word given its context)与Skip-gram(predicting the context given a word)；document embedding(将document变为vector——bag of word, 但是词汇的顺序代表了重要的含义，这是bag-of-word所不能顾及）[原理]

了解RNN（RNN实作）

* 直观感受一下RNN
* <https://www.data-blogger.com/2017/08/27/gru-implementation-tensorflow/> 利用tensorflow来创建一个GRU (RNN)
* 深入探讨RNN原理
* <https://www.youtube.com/watch?v=xCGidAeyS4M> 李宏毅讲RNN，重点讲解Long Short-term Memory(LSTM)[原理]
* <https://morvanzhou.github.io/tutorials/machine-learning/ML-intro/2-4-LSTM/> 莫凡讲解LSTM RNN
* 应用
* <http://frankchen.xyz/2017/12/18/How-to-Use-Word-Embedding-Layers-for-Deep-Learning-with-Keras/> 训练模型的建造——利用GLove这种单字嵌入方法对文本进行情感分类
* <https://machinelearningmastery.com/use-word-embedding-layers-deep-learning-keras/> 这个跟前一个连结如出一辙，他们都没有运用RNN，进行了二元文本情感分类

延伸

* <https://www.kaggle.com/rhodiumbeng/classifying-multi-label-comments-0-9741-lb> 多标签问题

￼